

REMARKS

Claims 1-20, 35-53, and 56-68 are pending in this application. Claims 1, 11, 35 and 42 are the independent claims. Claims 21-34, 54, and 55 are cancelled. New dependent claims 61-68 are added. Claim 50 is amended to add a missing period. Reconsideration is respectfully requested in view of the following remarks.

All of the rejections are based primarily upon U.S. Patent 6,976,262 to Davis *et al.* (Davis). Davis lacks the client-side object instantiation described in the pending claims. The client (Application Computer 150 including Client API 156 in FIG. 2A) has only basic tools to query the CIM Object Manager 20 on the server side (Computer System 110 in FIG. 2A). The boundary between client-side and server-side is illustrated in FIG. 2A of Davis as the dotted line.

The client in Davis does not know anything about the objects to be instantiated. A lean interface is employed to communicate with the CIM Object Manager 20. The client connects to the server to obtain an object model, which is instantiated by the CIM Object Manager 20, rather than the Client API 156. *See, e.g.*, FIG. 8 of Davis, and discussion of "Object Manager Execution" in columns 9 and 10. For example:

Preferably, application 32 invokes a method within Client API 156 which creates an instance of application 32 within object manager 20. Application 32 passes to the method a host name, a namespace, a user name, a password, and the protocol by which it is desirable to communicate with host computer system 110. Any suitable network protocol may be identified such as RMI, XML/HTTP or DCOM.
-- Davis column 9, lines 54-60 (emphasis added).

The client must query the the CIM Repository 130 via the CIM Object Manager 20 to find out what objects are available and how to access them. This open-system methodology in Davis allows dynamic changes to the CIM objects without any updating to the Client API 156, simplifying the addition of more models. The query objects in Davis have properties, but the structure and interface for the database objects reside on the server. This simplifies the client-side interface, but requires a communication channel to the server to instantiate an object and requires querying for content.

In comparison, referring to claims 1, 11, 35, and 42, the company object is instantiated on the client system. The client does not necessarily have to access the server to instantiate the object. Rather, the client need only connect to the server to act on server data.

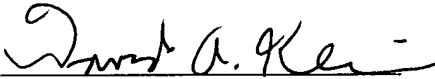
Based on these remarks, reconsideration of the rejections based on Davis is respectfully requested.

Applicant authorizes the Commissioner to charge any fees determined to be due with the exception of the issue fee and to credit any overpayment to Deposit Account No. 11-0600.

The Examiner is invited to contact the undersigned at (202) 220-4209 to discuss any matter concerning this application.

Respectfully submitted,
KENYON & KENYON

Dated: June 30, 2006


David A. Klein
Reg. No. 46,835

Kenyon & Kenyon
1500 K Street, N.W.
Suite 700
Washington, D.C. 20005
Tel: (202) 220-4200
Fax: (202) 220-4201